

**Department of Computing**

**BS Computer Science**

**NASTP Institute of Information Technology**

**Submitted By:** Maisum abbas (023)

**Submitted To:** Sir Ibaad Babar

**Semester:** II

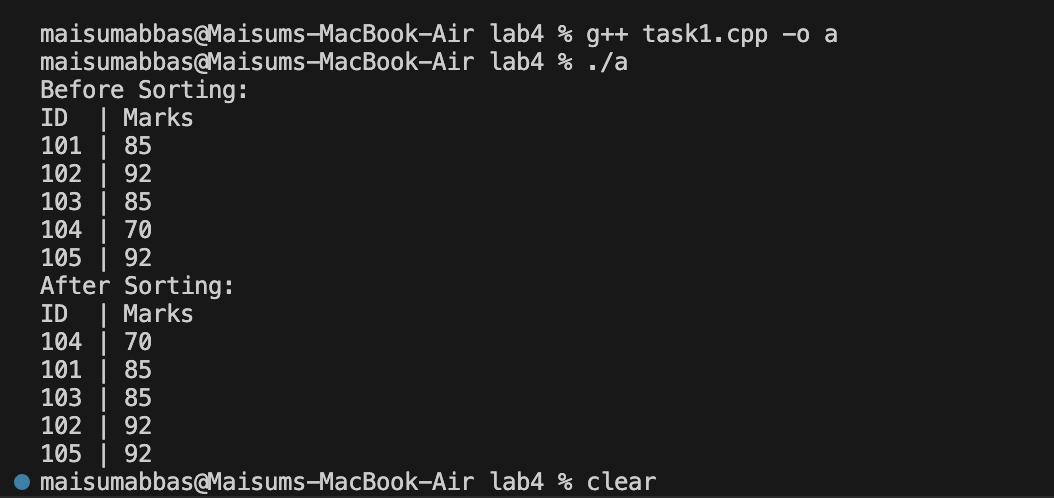
**Session:** Fall - 2024

**Subject:** Data Structures

**Date of submission:** 25 Oct 2025

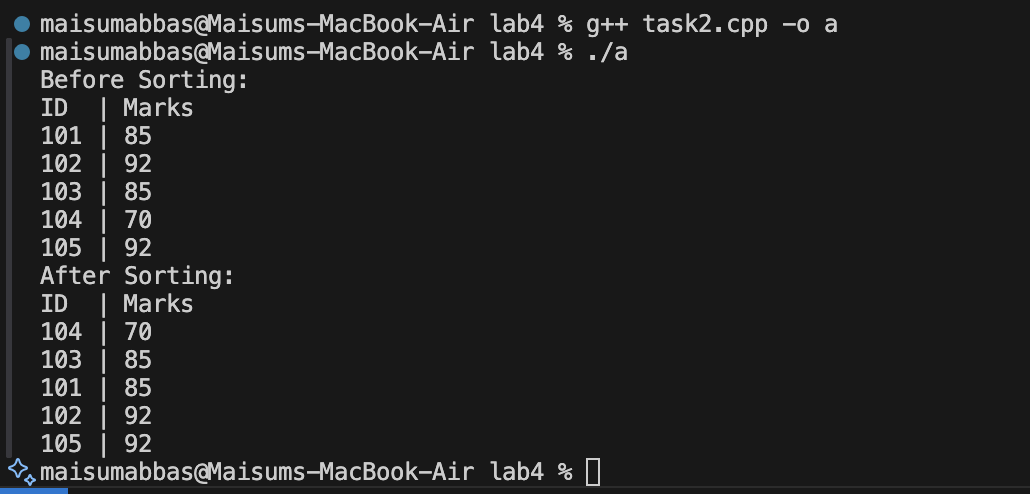
**Lab Report 4**

**Task1:**  
sorted the object attribute using merge sorting algorithm



**Task2:**

Sorted the object attribute using the quick sorting algorithm



**Task3:**

A sorting algorithm is considered stable if it preserves the original relative order of elements that have equal keys (or sorting values).

In this problem, the key is the student's marks. This means that if two students have the same marks, a stable sort will guarantee that they appear in the final sorted list in the same relative order as they were in the original list. An unstable sort makes no such guarantee.

Merge Sort maintained the original relative order for all pairs of equal-key elements, Merge Sort is stable. This is a well-known property of Merge Sort, achieved by its merge function, which explicitly prefers elements from the "left" (earlier) array when values are equal.

Quick Sort failed to maintain the original relative order (in this case, for both pairs), Quick Sort is unstable. This happens because its partition logic swaps elements over long distances based only on their comparison to a pivot, without regard for their original order relative to other equal elements.